

379  
N81  
No. 5593

SOFTWARE AND HARDWARE INTERFACE OF A VOTRAX TERMINAL  
FOR THE FAIRCHILD F24 COMPUTER

THESIS

Presented to the Graduate Council of the  
North Texas State University in Partial  
Fulfillment of the Requirements

For the Degree of

MASTER OF SCIENCE

By

Chun Hsiang Wu, M. S.

Denton, Texas

May, 1979

Wu, Chun Hsiang, Software and Hardware Interface of a VOTRAX Terminal for the Fairchild F24 Computer, Master of Science (Computer Sciences), May, 1979, 41 pp., 2 tables, 8 illustrations, 2 appendices, bibliography, 14 titles.

VOTRAX is a commercially available voice synthesizer for use with a digital computer. This thesis describes the design and implementation of a VOTRAX terminal for use with the Fairchild F24 computer.

Chapters of the thesis consider the audio response technology, some characteristics of Phonetic English Speech, configuration of hardware, and describe the PHONO computer program which was developed. The last chapter discusses the advantages of the VOTRAX voice synthesizer and proposes a future version of the system with a time-sharing host computer.

## TABLE OF CONTENTS

	Page
LIST OF TABLES . . . . .	v
LIST OF ILLUSTRATIONS . . . . .	vi
 Chapter	
I. INTRODUCTION . . . . .	1
Purpose of the Study	
II. PHONETIC ENGLISH SPEECH . . . . .	4
III. HARDWARE CONFIGURATION . . . . .	8
General Description of Fairchild F24 Computer	
General Description of VOTRAX	
I/O Characteristics of VOTRAX	
Interface VOTRAX with Fairchild F24 Computer	
IV. SOFTWARE CONFIGURATION . . . . .	16
Source	
Vocabulary	
Translation Algorithm	
V. THE OPERATION OF THE PHONO PROGRAM . . . . .	26
Loading	
Restart	
PHONO Program Commands and Examples	
PHONO Program Error Messages	
VI. DISCUSSION . . . . .	34
APPENDIX A . . . . .	38
APPENDIX B . . . . .	39
BIBLIOGRAPHY . . . . .	40

## LIST OF TABLES

Table	Page
I. VOTRAX Phonemes and Usage . . . . .	4
II. Glides in English and VOTRAX Phoneme Equivalents . . . . .	5

## LIST OF ILLUSTRATIONS

Figure	Page
1. Hardware Configuration . . . . .	8
2. Typical F24 Memory Reference Instruction Format . . . . .	10
3. F24 Select Peripheral Unit Instruction Format . . . . .	10
4. Parallel TTL I/O Interface of VOTRAX . . . . .	12
5. Main System Flow . . . . .	17
6. Vocabulary Table Element . . . . .	21
7. The Linked Relation between a Vocabulary and the Corresponding Phonetic Codes Definition . . . . .	23
8. Proposed New System Flow . . . . .	36

## CHAPTER I

### INTRODUCTION

Audio response is in many instances an ideal component in man-machine communication (1). This approach can be used in combination with other means of presentation to provide an oral explanation of text (2) or music\* to keep up the students' interest.

The attractive features of the audio response in man-machine communication include the following:

1. Audio response is a natural way for humans to receive communications from others.

2. Audio messages can be received without interrupting the use of the manual or visual channels, in motion, or in total darkness.

3. Any number of listeners can receive the same audio message simultaneously.

There are some eighteen American suppliers (3) of voice output equipment. The VOTRAX\*\* Voice Synthesizer (VOTRAX)

---

\*The computer aided instruction music learning system currently under development in the Department of Computer Sciences, North Texas State University, Denton, Texas.

\*\*Trade-mark of Vocal Interface Division of the Federal Screw Works, Troy, Michigan.

appeared to be ideal for Computer Aided Instruction (CAI) Systems because it is relatively inexpensive and digitally controllable. VOTRAX has an essentially unlimited English vocabulary, in contrast to other devices, which usually select only prerecorded words. The basis for the vocabulary of VOTRAX is that it accepts sequence of digital phoneme commands (a phoneme is a basic unit of speech) and translates these commands into corresponding phonetic audio message.

Algorithms have been developed (1) which convert English text directly into the corresponding phonemes to drive the synthesizer. By using an optical reading device, printed text may be scanned and automatically converted into speech. It is speculated that this system could then be used as an aid to the blind (2).

#### Purpose of the Study

This thesis describes an effort to program the VOTRAX interactively under control of a Fairchild F24 mini-computer. The PHONO program, written in F24 Assembler Language, provides for user English instruction commands of the VOTRAX. The primary design problem was to allow users to synthesize speech and to develop new vocabularies which can be stored in the vocabulary table of the computer and used upon demand.

## CHAPTER BIBLIOGRAPHY

1. Flanagan, J. L., Speech Analysis Synthesis and Perception, Berlin-Heidleberg, Springer-Verlag, 1972.
2. Ehrlich, Michael D., The VOTRAX Voice Synthesizer as an Aid for the Blind, Troy, Michigan, Vocal Interface Division of the Federal Screw Works, 1974.
3. All About Voice Response, Datapro 70, September, 1974.



## CHAPTER II

### PHONETIC ENGLISH SPEECH

The material for this chapter was derived from Reference 1, for the convenience of the reader's understanding and reference.

Table I lists all the VOTRAX phonemes (1) and their probable usage.

TABLE I  
VOTRAX PHONEMES AND USAGE

Phoneme Name	Usage	Phoneme Name	Usage	Phoneme Name	Usage
A, AY	eight	A1	eighty	AE	cat
AE1	antenna	AH	car	AH, IY	nine
AH1	connect	AW	law	AW1	fault
B	bed	T, CH	catch	D	do
D, J	edge	EH	ten	EH, EHH	Ed
EH1, EH2	seven	ER	weather	F	for
G	get	H	hello	I	six
I, IH	sin	I1	inept	I2	static
IE	zero	IE, E	three	IH	station
IU, U	two	K	came	L	hello
M	mile	N	nine	NG	ring
O	no	O1	hello	O2	notice
OO	book	OO, OOH	bush	OO1	good
P	penny	Q	quick	R	three
S	six	SH	show	T	ten
TH	three	THV	then	U1	thorough
UH	but	UH1	uncle	UH2	stirrup
UH3	apple	V	seven	W	won
Y	sixty	Y1	yes	Z	zero
ZH	azure				

Vowel sounds which are considered by the unsophisticated to be basic sounds can be in reality glides of vowels in combination, such as long A, short A, long E, short E, long I, short I, long O, and long U. Other sounds are also combination of two or more basic sounds, such as combination of T and CH in "CATCH". The other numbered phonemes are lesser stressed versions of those listed in Table I.

It is important to know the many glides found in English and how to generate them with the VOTRAX equivalent.

Table II shows some examples.

TABLE II  
GLIDES IN ENGLISH AND VOTRAX PHONEME EQUIVALENTS

Common Phoneme Name	VOTRAX Phoneme Equivalent
long A in name	A AY
short A in hand	AE EHH
AW in raw	UH AW
long E in three	IE E or IE Y
short E in Ed	EH EHH
long I in high	AH IY
short I in is	I IH
long O in low	UH O
OW in cow, OU in out	AH O
OI in noise	O I2 Y1
short O in hop	UH AH
long U in tune	IU U
double O in took	OOH OO
double O sound in bush	OO OOH
J, DG in edge	D J
G in George	D J
CH in chair	T CH
NK in thank	NG K
NG in engage	N G
NG in engine	N D J
beginning Y like in yes	Y1

With all vowels, appropriate stress must be determined and the corresponding phoneme chosen. The word "eight", for example, is spelled A AY T while the word "eighty" is spelled A1 AY TY. The A1 in place of A shortens the syllable of "eighty" to correspond with our natural pronunciation of that word. As a general rule, longer words have short syllables and require more higher numbered, shorter vowel commands.

The more elaborate synthetic English speech rules (1) you follow, the more intelligible speech you get.

## CHAPTER BIBLIOGRAPHY

1. McIlroy, M. D., Synthetic English Speech by Rule,  
Murray Hill, New Jersey, Bell Telephone Laboratories,  
March, 1974.

## CHAPTER III

### HARDWARE CONFIGURATION

The hardware configuration of this VOTRAX Voice Synthesizer terminal is shown in Figure 1.

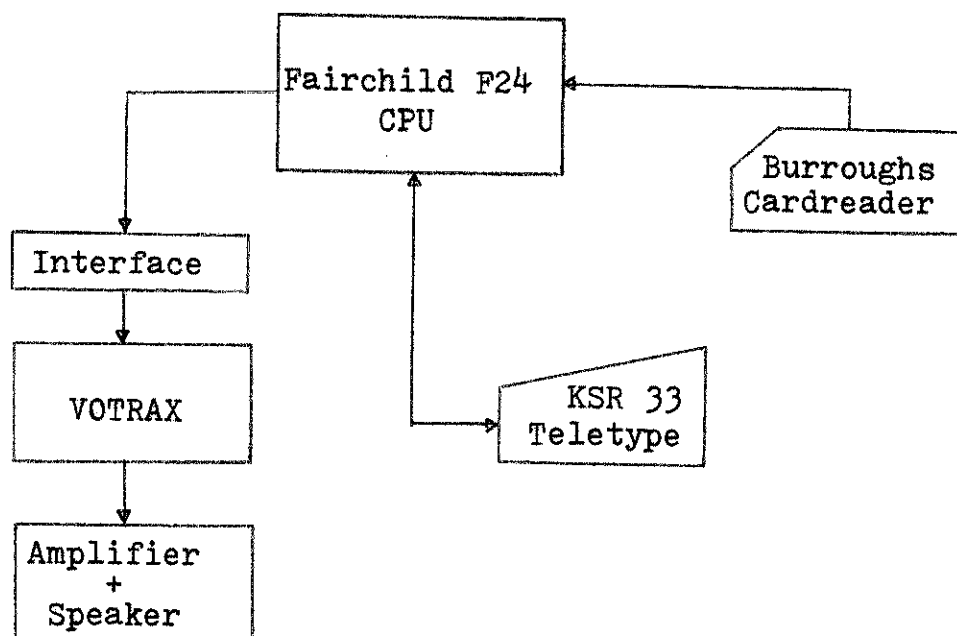


Fig. 1--Hardware configuration.

The VOTRAX is connected to a Fairchild F24 computer which has 4196 words of 24 bits. The interface was designed (1) as a North Texas State University Computer Sciences course project.

## General Description of Fairchild F24 Computer

The Fairchild F24 mini-computer used on this project is a high-speed general purpose computer.

Donovan (2) has stated general rules for describing a computer; listed below are the characteristics of Fairchild F24 computer, following Donovan's outline (3).

### Memory

The basic unit is a word of 24 bits (or bytes) of information. The unit size of core memory is 4k words and is expandable in modules of 4k up to 16k words. All 16k words are directly addressable. The complete memory cycle requires 1.6 microseconds. Memory address zero is reserved for the interrupt return address storage. Memory locations 1 through 63 are reserved for storage of indirect addresses for the external interrupt system.

### Registers

The Fairchild F24 has 1 accumulator and 1 accumulator extension register, each consisting of 24 bits, and 8 index registers consisting of 14 bits each.

### Data

All data and instructions are stored as sequences of binary ones and zeros.

### Instructions

The Fairchild F24 computer has arithmetic, logical, control, and special interrupt instructions. Typical instruction formats are as shown in Figures 2 and 3.

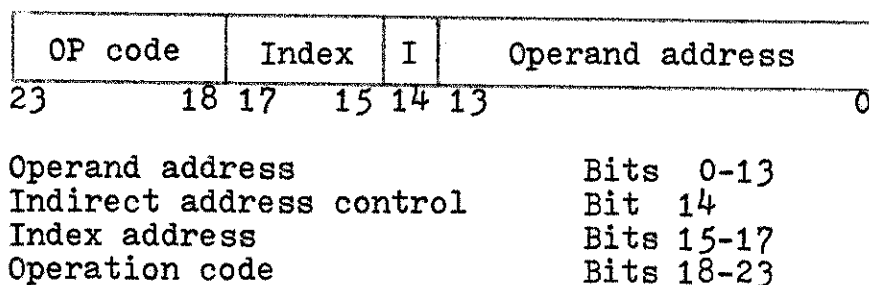


Fig. 2--Typical F24 memory reference instruction format.

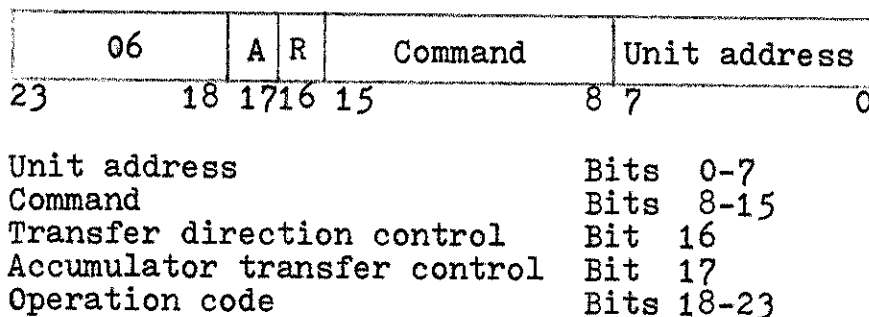


Fig. 3--F24 SPU (Select Peripheral Unit) instruction format.

### Special Features

The F24 has 16 external interrupt channels with up to 63 distinct interrupt locations. There are also 8 programmable state-switch bits and 6 console switches, in addition to a large number of switches used for initialization and diagnosis of the CPU.

### General Description of the VOTRAX

The VOTRAX (Model #5-2219-001-00) can be described as an electronic simulator of the human vocal system. Its audio output can be transmitted over speaker systems or telephone lines. The manufacturer states that by sending an appropriate string of phonemes, any word in the English language can be uttered. In practice, however, it was found that the quality of speech production of the model used is not adequate for many purposes.

The characteristics of a phoneme are selected by the value of a parallel 8-bit binary word. Two of the 8-bits select one of four inflection levels, the other 6 bits select one of 64 phonemes.

Under digital computer control, the vocabulary or a string of phonemes can be stored in computer memory or secondary storage and then transmitted to the VOTRAX for pronunciation.

### I/O Characteristics of the VOTRAX

The VOTRAX input and output logic signals are TTL compatible. Signal description (4) is shown in Figure 4.

The logic signal, Input Set, enables the VOTRAX to receive data upon the initiation of the data strobe. Each strobe enters one phoneme and its inflection into an 80-character shift register buffer. Input Set is complemented by an Output Set pulse. When in the output mode, no input



data is accepted. Inversely, no output can be performed while in the input mode.

Pin no.    Function

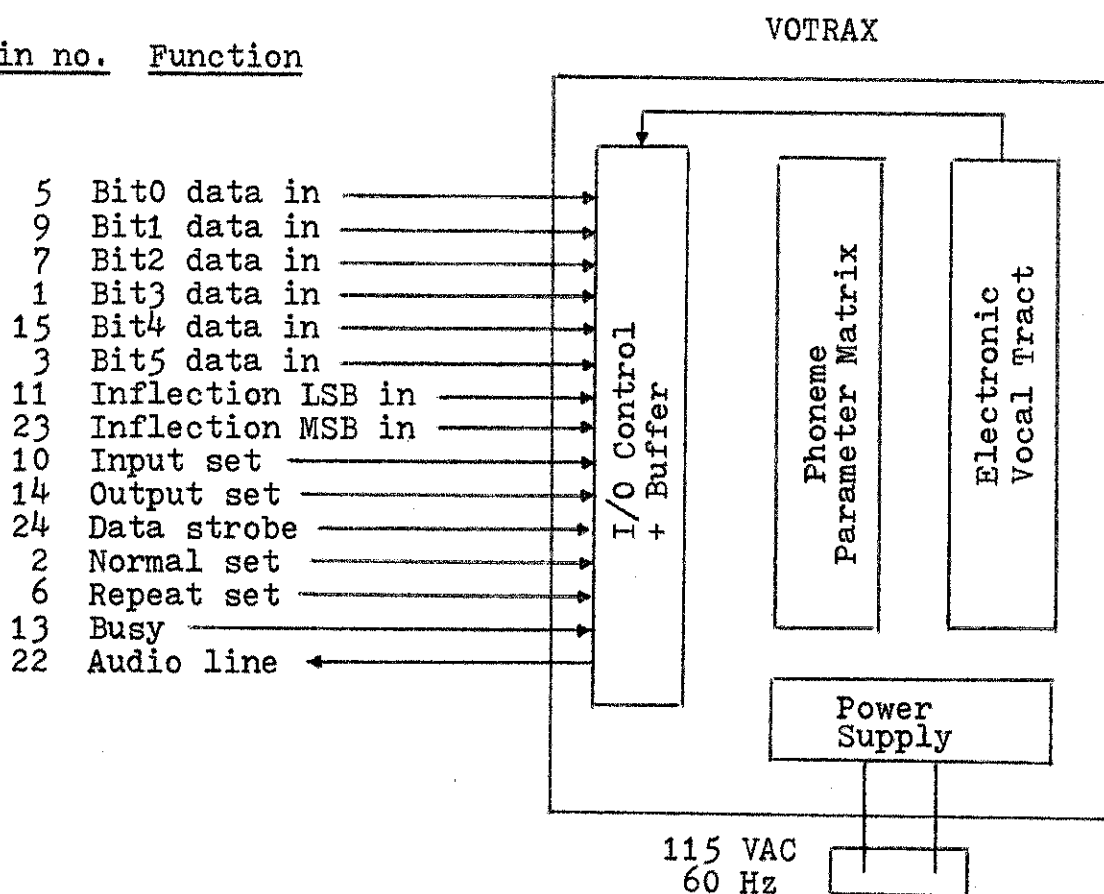


Fig. 4--Parallel TTL I/O interface.

Normal Set is the regular application of VOTRAX.

Repeat Set is for repeating a message.

For maximum input (80 bytes) of phoneme data, a pause phoneme (PA0) must be entered as the first data. When a phoneme byte other than null (unsounded, 0 time code) reaches the end of the VOTRAX buffer, the VOTRAX speaks out the first phoneme even though it is still in the input mode.

When in the output mode, the busy signal is set and VOTRAX speaks out the phoneme command in sequential order. The rate of talking can be adjusted manually through a speech rate controller on the front panel of the VOTRAX.

When VOTRAX is in the output mode and the buffer is empty, the busy indicator automatically is reset after 768 milliseconds. The VOTRAX continues to output nulls as long as it is in the output mode.

#### Interface VOTRAX with Fairchild F24 computer

Execution of the Select Peripheral Unit (SPU) instruction (5) of the F24-Fairchild computer presents the least significant 18 bits of the SPU instruction word to the system peripherals on the accumulator bus during the time the peripheral select (PS) synchronization signal is energized. At  $T_4$  of SPU, the least significant 18 bits of the command register are gated to the accumulator bus. Each peripheral device receiving the PS signal decodes the least significant 8 bits of the accumulator bus to determine whether or not these bits contains its own unit code. If the SPU instruction is of the form requiring an input or output accumulator transfer, the data on the accumulator bus at  $T_1$  time of the SPU is loaded into the accumulator, or the contents of the accumulator is gated onto the accumulator bus at  $T_1$ , depending on the values of bits 17 and 16 of the command. If bit 17 is set in the SPU command there is an

accumulator transfer made at  $T_1$  of the SPU. If bit 16 is not set, the transfer is from the accumulator to the addressed peripheral via the accumulator bus.

From the logic design drawing (see Appendix B) of the interface device and the SPU instruction format as shown in Figure 3, the SPU instruction bit pattern for the VOTRAX is as follows:

1. The device code of VOTRAX ( $371_8$ ) is decoded from the SPU command, bits 0-7, at  $T_4$  time.
2. Bit 10 of the SPU must be "1" to enable Input/Output. Bit 12 of the SPU must be "1" to enable Normal/Repeat.
3. Since a "low" voltage to the Input, Output, Normal, and Repeat sets of VOTRAX means "true"; therefore, if we want the VOTRAX to be in the Output-Repeat mode, bit 9 and bit 11 must all be "0".
4. The data from accumulator bus is gated at  $T_1$  time for strobe-in.
5. The two D flip-flops serve as a two-bit shift register gating the device code to the data latching at  $T_1$  time.
6. The strobe-in as shown in Appendix B is "0" most of the time, but when it is the proper device code and bit 8 is "1" then there will be a pulse sent out and the latched data transferred into the VOTRAX buffer.

## CHAPTER BIBLIOGRAPHY

1. Smith, Russel, unpublished interface design, graduate student, Department of Computer Sciences, North Texas State University, Denton, Texas, 1975.
2. Donovan, John J., Systems Programming, NY, McGraw-Hill Book Co., 1972.
3. Scott, Dan W., "The Fairchild F24 Computer Systems Manual," unpublished manual, Department of Computer Sciences, North Texas State University, Denton, Texas, 1973.
4. Bulletin #ETB-5-0302-MD, Troy, Michigan, Vocal Interface Division of the Federal Screw Works, 1973.
5. Fairchild F24 Systems Manual (Hardware), Vol. 3 (3 volumes), California, Fairchild Instrumentation, 1970.

## CHAPTER IV

### SOFTWARE CONFIGURATION

The main programming effort was to convert words and sentences to phonetic images in a form acceptable by VOTRAX. The phonetic string is then transmitted from the computer memory to the VOTRAX buffer for speech generation.

The main system flow is shown in Figure 5.

#### Source

The input to the PHONO program (PHONO) is called the source. The source can be entered through Teletype, card reader, or paper tape reader.

The source is a string of characters. PHONO recognizes certain characters and certain groups of characters as different types of atoms. A syntactic element of the PHONO recognized by its specific class is called an atom. The atoms recognized are inflection levels, phonemes, words, and delimiters.

#### Inflection Levels

Four types of inflection levels are defined for the PHONO. These are:

- 1 - Lowest (shorter and softer amplitude)
- 2 - Low (short and soft amplitude)
- 3 - High (long and loud amplitude)
- 4 - Highest (longer and louder amplitude)

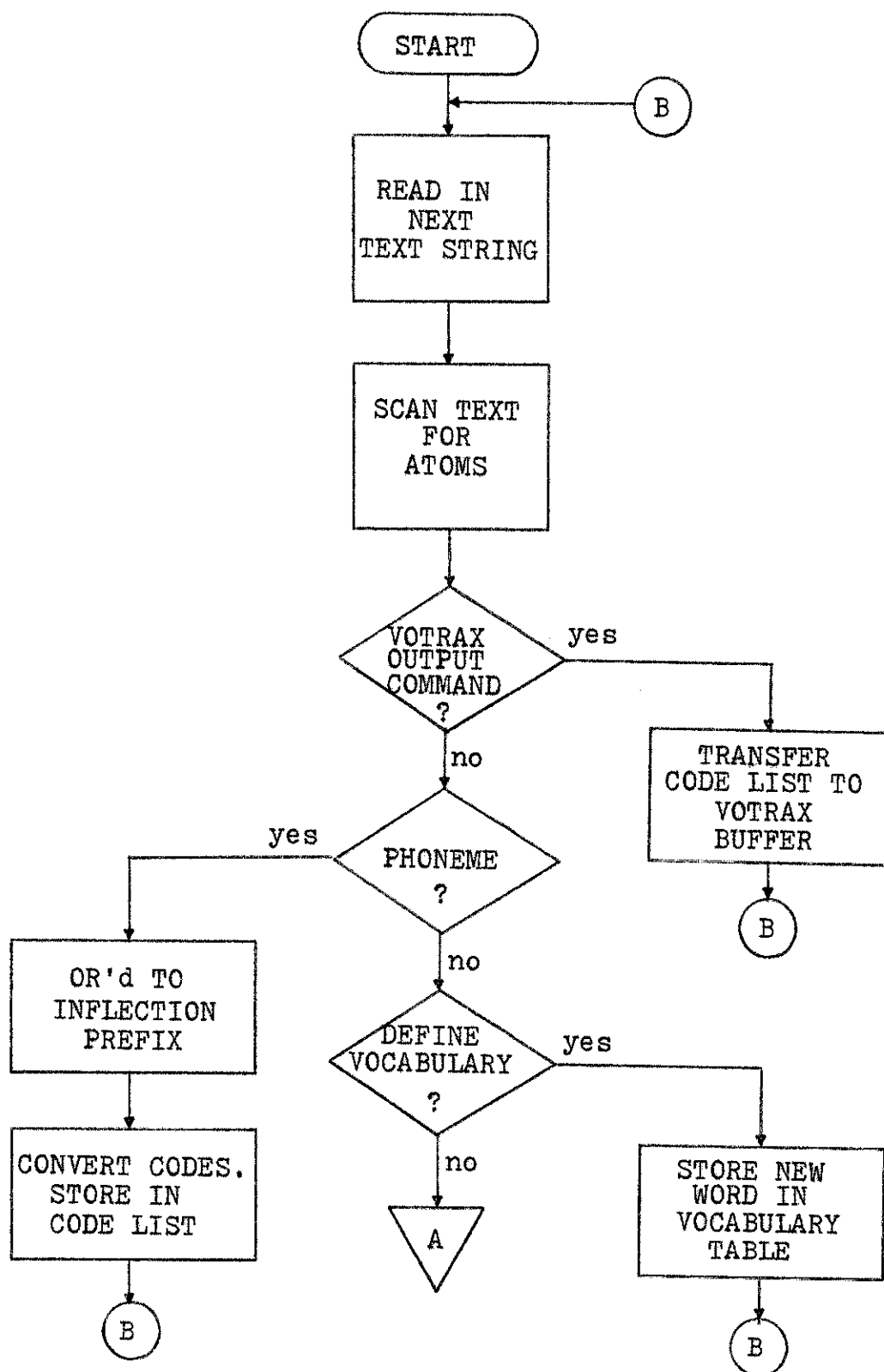


Fig. 5--Main system flow.

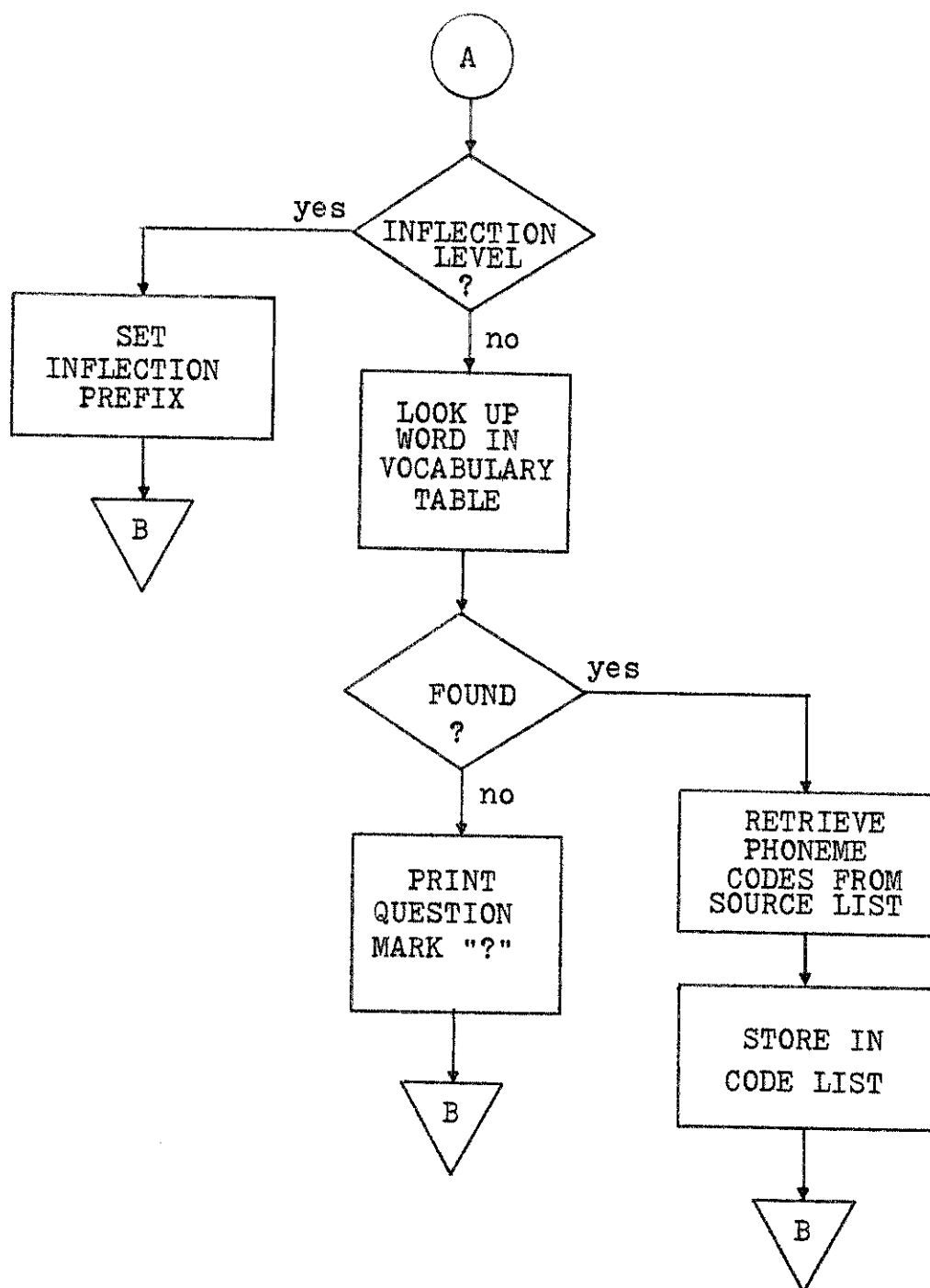


Fig. 5--Continued.

An inflection level is represented as an integer, having the range 1-4, followed by a phoneme.

### Phonemes

Sixty-three different phonemes are defined for the PHONO (see Appendix A). A phoneme has to be preceded by an inflection level or a slash "/". In the case of a slash, the phoneme assumes the default inflection, which is the inflection level of the previous phoneme.

### Words

A word for the PHONO has to be defined to be spoken. The phonemes of a word have to be in the PHONO vocabulary table. The construction of the vocabulary table is discussed later in the chapter.

### Delimiters

The delimiters are classified as commands and break characters. A command is a class of delimiters which is preceded by a dollar sign, "\$". The commands are:

\$	Repeat speak VOTRAX code, erase the code list.
\$S	Speak once from the code list.
\$R	Repeat speak, using the code list.
\$Q	Quit speech.
\$EC	Erase the VOTRAX code list.
\$ES	Erase the source list.
\$D	Go to program Memory Dump.
\$L	Go to program Loader.



The break characters serve the general function of separating phonemes and vocabularies from other phonemes and vocabularies. The break characters are:

b      b represents the class of spaces--a space, a comma, a period, a bar, a hyphen, or any number of combinations of them.

/      A slash indicates the beginning of a phoneme.

### Vocabulary

The result of defining the words to be used is called the vocabulary. The vocabulary consists of a collection (or set) of phrases and their associated words. A phrase consists of one or more words (keys) in a sequence. Example: 'I AM THE PRESIDENT'. The word identifier, which uniquely identifies a word, is referred to as the key of the word. A word consists of zero or more keys and/or zero or more basic constituents called atoms (or phonemes) in sequence. The vocabulary table is an ordered (sequential) list of elements. The information of a phrase definition consists of a set of atoms. The classes of atoms are:

1. Word increments
2. Keys
3. Terminators
4. Indices

### Word Increments

Byte zero of a vocabulary table elements has a number, called word increment, which specifies the number of consecutive computer words for a specific vocabulary in the table. A typical word increment is as shown in Figure 6.

The definition of the vocabulary "ANSWER", for example, takes 4 computer words; where "4" is the word increment for the vocabulary "ANSWER". The increment word serves as a secondary key (1) when searching a vocabulary in the vocabulary table; it identifies all those vocabularies which have a certain number of characters. Only when the word increments match is an actual vocabulary search performed. This technique eliminates the exhaustive comparison of character by character in the process of searching a vocabulary.

byte 0

N	A	4
E	W	S
377 <sub>8</sub>	377 <sub>8</sub>	R
index		
⋮		

Fig. 6--Vocabulary table element.

### Keys

Following the word increment in a vocabulary table element is a string of ASCII characters called a key. The A-N-S-W-E-R as shown in Figure 6, for example, is the key.

### Terminators

When a computer word is not fully occupied by the vocabulary, the rest of the computer word is filled with all "1"s, called terminators, as shown in Figure 6. The terminator is represented by an easily recognizable value, "377<sub>8</sub>", which can not be mistaken for an ASCII character.

### Indices

The last computer word of a vocabulary table element is an index into an index table. The index table has a list of the byte offsets into the source list (a set of the vocabulary phoneme code definitions) as shown in Figure 7.

### Translation Algorithm

The translation algorithm of the PHONO program is shown in the main system flow chart of Figure 5.

### Data Bases

Phoneme table.--Each entry has two computer words. The first word contains the ASCII phoneme name left justified. The second word is the corresponding VOTRAX phoneme code.

Vocabulary table.--A list of all the defined vocabularies.

Source list.--A set of the phonetic codes definitions for the vocabularies.

Code list.--A list of the generated phoneme code ready to be sent to the VOTRAX.

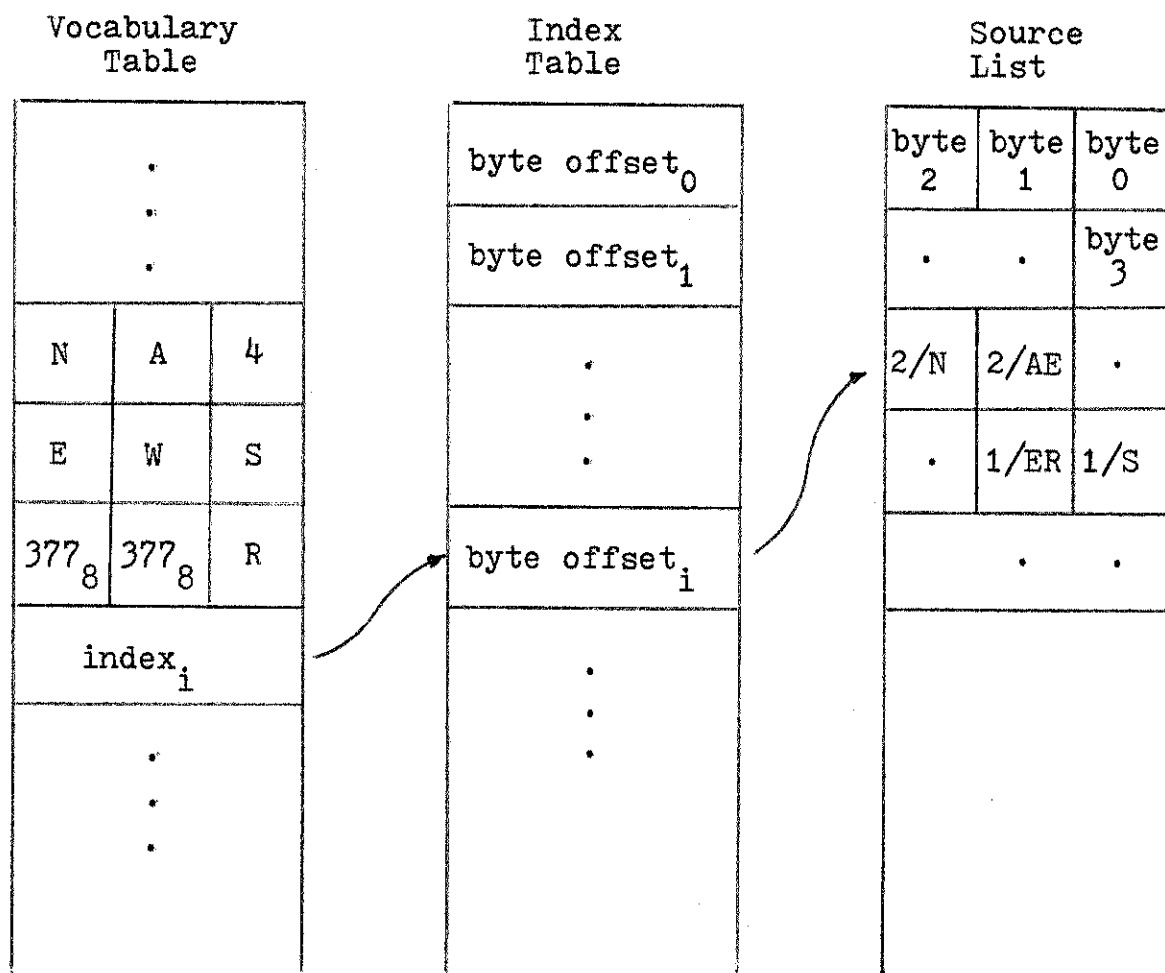


Fig. 7--The linked relation between a key and the corresponding phonetic codes or keys definition.

### Phoneme Codes Generation

Scanning determines the text elements. If an inflection level is detected, the inflection prefix is set to the corresponding value. When a phoneme is detected, it is used as a search argument into the phoneme table. If the search is successful, the associated octal code is OR'd to an inflection prefix for the derived sound; and then stored in the output phoneme code list first in first out (FIFO).

When a "word" is detected, it is used as a search argument into the vocabulary table. If the search is successful, the corresponding phonetic codes definition is retrieved from the source list and moved to the phoneme code list. If a PHONO command is detected, it is tested to see if it is a VOTRAX command. If it is a VOTRAX command, the output phoneme code list is transferred to the VOTRAX buffer for speech generation.

## CHAPTER BIBLIOGRAPHY

1. Knuth, Donald E., The Art of Computer Programming, Vol. 3 (3 volumes), California, Addison-Wesley Publishing Co., 1968.

## CHAPTER V

### THE OPERATION OF THE PHONO PROGRAM

#### Loading

The PHONO object program, which is in the form of punched-card output of the F24 Assembler (1) run on the IBM/360-50 in the University Computer Center, can be loaded into the F24 memory from the card reader.

The procedure is as follows:

1. Turn on the CPU and card reader power and set them in the STOP state.
2. Place the one-card Bootstrap Loader (2) in the card reader hopper.
3. Turn card reader VALIDITY off.
4. Depress card reader RESET and START.
5. Press CPU RESET and LOAD.
6. Set command register switches to  $01000100_8$ .
7. Press RESET and LDCR.
8. Press START up and down.
9. Switch the Teletype power switch to ONLINE (counter clockwise).
10. Place the Memory Dump Program (3) in the Teletype paper tape reader.
11. Switch the Teletype paper tape reader to RUN.

The Memory Dump Program will be read into core.

12. Place the Card Loader Program (4) in the Teletype paper tape reader.

13. Switch the Teletype paper tape reader to RUN.  
The Card Loader Program will be read into core.

14. STOP the CPU.

15. Place the PHONO object program, which should be followed by a blank card, in the card reader hopper.

16. Turn card reader VALIDITY off.

17. Depress card reader RESET and START.

18. Set command register switches to  $01000300_8$ .

19. Press RESET and LDCR.

20. Press START up and down. The PHONO program will be read into memory and gain control.

21. The PHONO program will then prompt a user response by typing

>

on the Teletype.

Commands can be input from the Teletype input, the paper tape reader, or the card reader.

State Switch 13 (Console Switch 3) is used to control the trace (VOTRAX octal code) of the VOTRAX output on the Teletype; Console Switch 3, if up, is equivalent to TRACEON. State Switch (Console Switch 5) is used to control the echo of the input on the Teletype; Console Switch 5, if up, is equivalent to NOECHO.



### Restart

The PHONO program can be restarted at any time by pressing STOP, setting the command register switches to 01001001<sub>8</sub> and then pressing LDCR and START. The PHONO program assumes operation as in the state it is first started except it does not erase the vocabulary table and the phoneme code list.

### PHONO Commands and Examples

Blank group.--These characters are used to separate other groups, such as words. Members of the group are all equivalent. The group consists of the blank space, return, and line feed. A blank which immediately follows a blank is ignored.

Back arrow (<) or underline.--Causes the last character typed in to be ignored, unless the last character was a blank. Examples:

"UV<H2" is the same as "UH2".

"CONP<<MPUTER" is the same as "COMPUTER".

Digit, 1 to 4.--Sets the inflection level. A question mark (?) is printed and field ignored if the digit is outside the inflection level range (1 to 4). Example:

2/AE /N 1/S /ER

A slash "/" is printed immediately after the digit by the PHONO program. "AE" above has an higher inflection level

than the "S". The phoneme assumes the previous inflection level when no inflection level (b/phoneme) is specified. In this case, "N" has the same inflection level as "AE".

Phoneme.--Specifies one of the phonemes. A question mark (?) is printed and field ignored if the phoneme entered is illegal. Example:

"AE1" is translated to the VOTRAX code value of 57<sub>8</sub>.

- (Hyphen).--Phoneme to speak a brief pause between or within words (Hyphens may not be used in spelling or punctuation). Example:

PRESIDENT NIXON NOMINATED A NEW VICE-PRESIDENT.

, (Comma).--Phoneme to pause between phrases. The comma pause is longer than the hyphen pause (Commas may not be used in spelling). Example:

TODAY, PRESIDENT NIXON NOMINATED A NEW VICE-PRESIDENT.

. (Period).--Phoneme to pause between sentences. The period pause is longer than the comma pause (Periods may not be used in spelling). Example:

TODAY, PRESIDENT NIXON NOMINATED A NEW VICE-PRESIDENT.  
HE IS GERALD FORD OF MICHIGAN.

" (Double quote).--Defines vocabulary (enter a new source in the vocabulary table). Example:

"

A response (left parenthesis "(") is typed on the Tele-

type, and then the user can enter the vocabulary ending with a right parenthesis ")".

"(ANSWER)

A response (colon ":") is typed immediately following the ")"; at this time the user can enter the definition of the vocabulary ending with a carriage return or a double quote. The vocabulary definition can be a combination of phonemes and pre-defined keys.

1. "(ANSWER): 2/AE /N 1/S /ER

2. "(ANSWERYES): ANSWER 2/Y1 /A 1/AY /S"

The vocabulary can be redefined (erase the previous definition), which enables the user to modify the pronunciation of the vocabulary.

3. "(ANSWER): 3/AE 2/N 1/S /ER

( (Left parenthesis) )--Same as "" (double quote), except no first response (left parenthesis "(") is typed on the Teletype.

\$Q--Quit speech. Example:

\$Q

"UIET" is printed immediately following "\$Q". Nulls are sent to VOTRAX, hence the VOTRAX quits speaking.

\$S--Speak once from the phoneme code list. Examples:

1. 2/AE /N 1/S /ER \$S

"ANSWER" is uttered once.

## 2. ANSWER \$S

"ANSWER" is uttered once. "ANSWER" must be pre-defined.

## 3. \$S

What is in the phoneme code list is uttered once.

\$R.--Speak repeatedly from the phoneme code list.

Similiar to "\$S"; the only difference being that this command repeats the uttering.

\$blank.--Repeat speaking the phoneme code, then erase the phoneme codes list. Example:

2/AE /N 1/S /ER \$

"ANSWER" is uttered repeatedly. Since the phoneme code list has been erased, no sound would be output if a "\$S" command were next typed in.

\$EC.--Erase the phoneme code list.

\$ES.--Erase the source list.

\$L.--Go to program Loader. Example:

\$L

"OAD" is printed immediately following "\$L". Control is transfered to the Bootstrap Loader (location 100<sub>8</sub>), hence PHONO enables the user to load and execute a new program.

\$D.--Go to the program Memory Dump in order to diagnose

programming errors. Example:

\$D

"UMP" is printed immediately follows "\$D". The starting dump address is entered from the Teletype. The memory dump may be aborted at any moment by striking any key on the Teletype.

### Error Messages

Too many symbols.--If the user tries to define a vocabulary which has more than 20 characters, the above error message will be typed on the Teletype.

Too many source words.--When the vocabulary table is full, the above error message will be typed on the Teletype and the CPU halted at location 1001<sub>8</sub>.

In this case, the user can press START up and down and restart again. Since the vocabulary table is full, the user may either issue a "\$ES" command to erase the vocabulary table or continue without trying to define more vocabulary.

## CHAPTER BIBLIOGRAPHY

1. Scott, Dan W., unpublished F24 simulator on the IBM 360, Department of Computer Sciences, North Texas State University, Denton, Texas.
2. Scott, Dan W., unpublished one-card bootstrap loader, Department of Computer Sciences, North Texas State University, Denton, Texas.
3. Scott, Dan W., unpublished memory dump program, Department of Computer Sciences, North Texas State University, Denton, Texas.
4. Scott, Dan W., unpublished card loader program, Department of Computer Sciences, North Texas State University, Denton, Texas.

## CHAPTER VI

### DISCUSSION

The VOTRAX terminal can be implemented on numerous applications ranging from computer aided instruction systems to on-line student information inquiry systems (1).

The present system is based on the available Fairchild F24 mini-computer in the Department of Computer Sciences, which has only 4k memory of 24 bits/word. The size of the vocabulary is limited to a few hundred words. Suppose we use the time-sharing computer (HP-2000) in the University Computer Center, which has available mass secondary storage, as a host computer remotely accessed by the F24; the vocabulary size is then virtually unlimited. The proposed new system flow is shown in Figure 8.

The vocabulary file can be read into the F24 computer core memory; then the file transferred to host CPU memory by a program written in BASIC. After sorting by merging (2), the vocabulary file just read in will be merged into the previous vocabulary file, hence eliminating re-constructing the lexicographical (3) vocabulary file.

The most-often used words can be kept in the F24 mini-computer for speed considerations. As for the vocabulary search of those stored in the host time-sharing computer

secondary storage, a record can be brought into the F24 memory by retrieving on secondary keys. A binary search (3) is then performed and phonemes sent to VOTRAX for speech generation.

By utilizing a host computer the range of applications would be greatly extended, particularly for spoken output to file inquiries. If an outside telephone could call the F24, connected to the host computer, then a person anywhere in the world could receive the answer inquiries relating to large data bases.



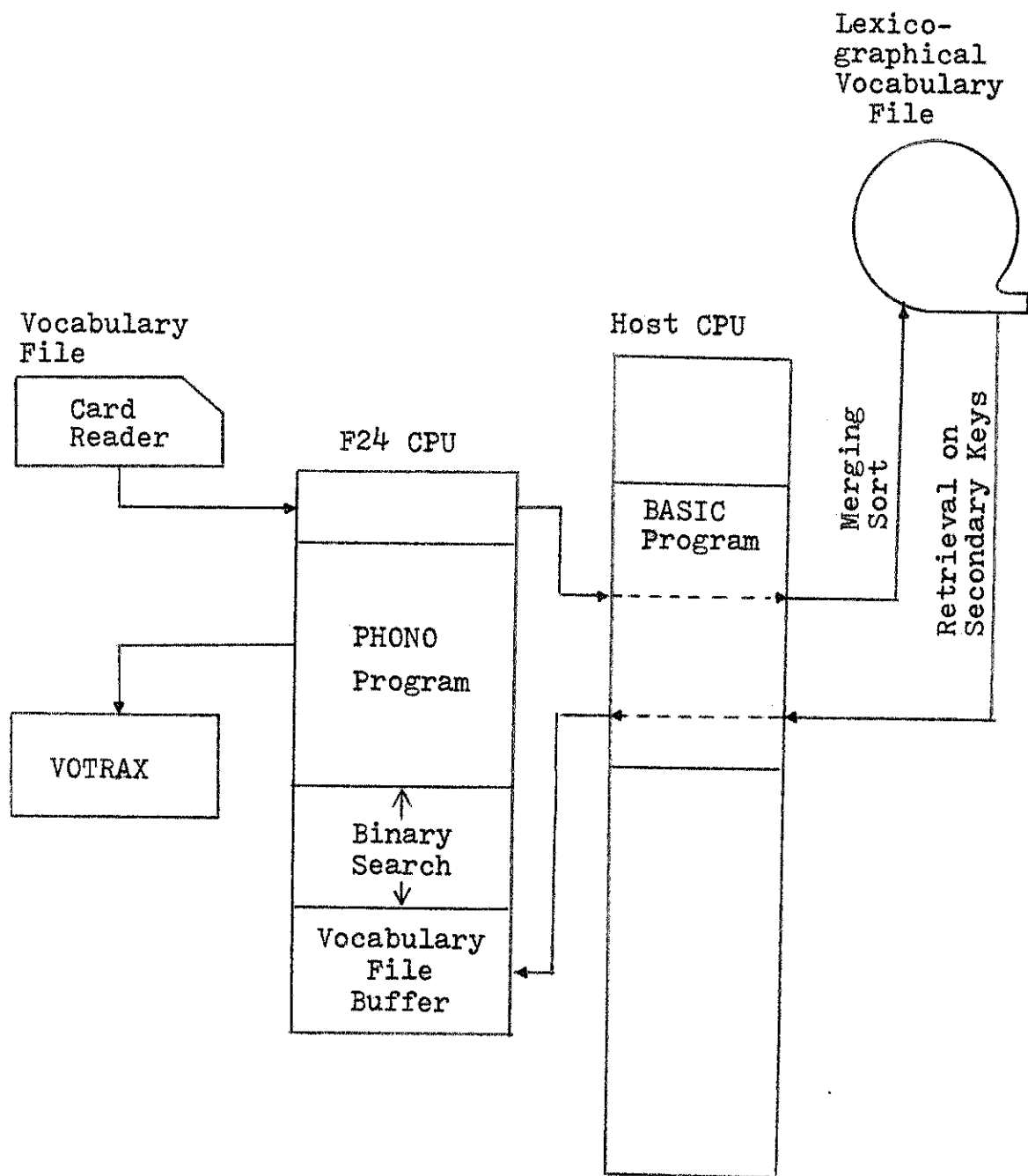


Fig. 8--Proposed new system flow.

## CHAPTER BIBLIOGRAPHY

1. Clements, David T., Voice Output for Student Information Inquiry, Costa Mesa, California, Coast Community College District, 1974.
2. Knuth, Donald E., The Art of Computer Programming, Vol. 3 (3 volumes), California, Addison-Wesley Publishing Co., 1968.
3. Donovan, John J., Systems Programming, NY, McGraw-Hill Book Co., 1972.

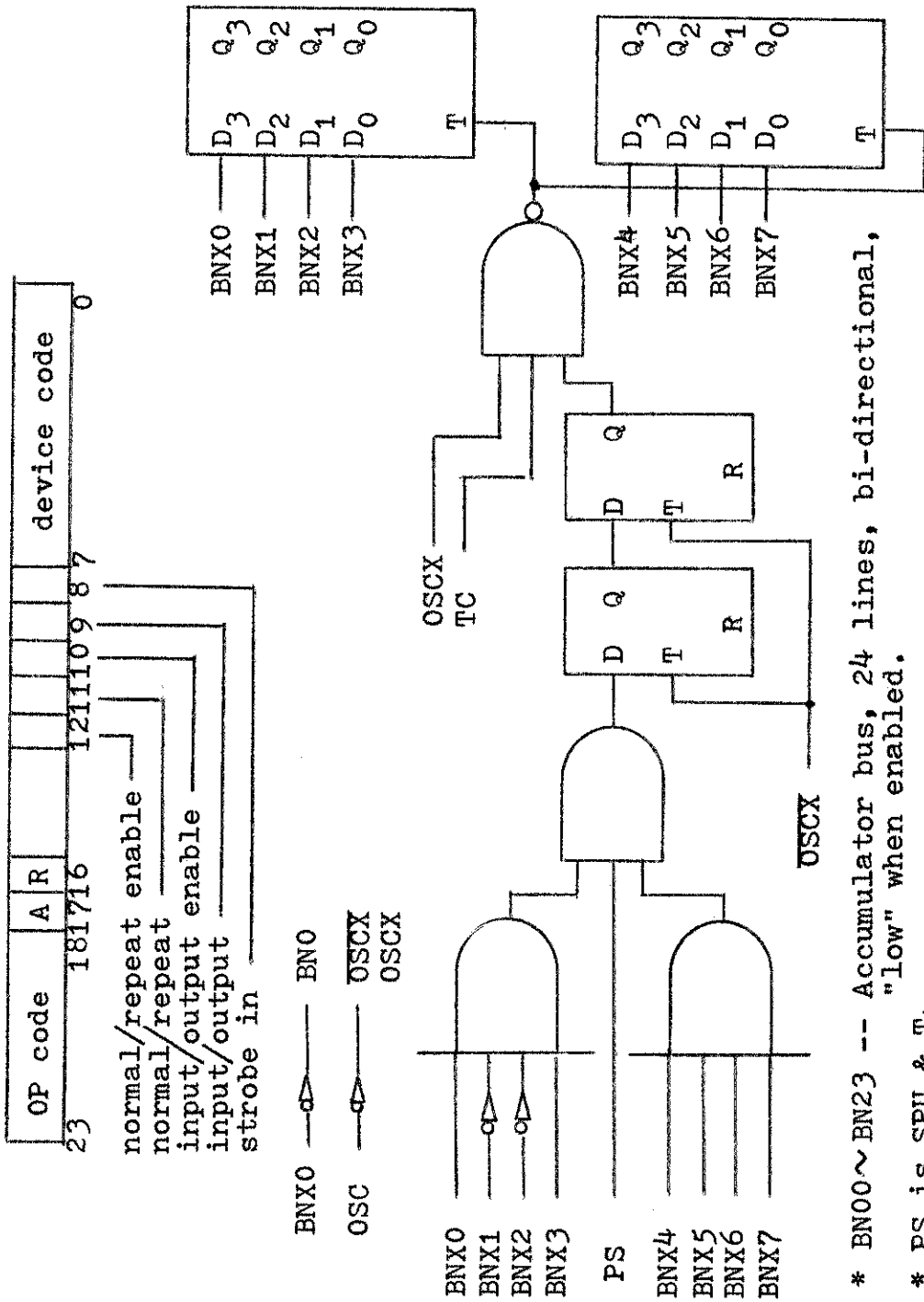
# APPENDIX A

## VOTRAX PHONETIC CODES

<u>Phoneme</u>	<u>Octal</u>	<u>Phoneme</u>	<u>Octal</u>	<u>Phoneme</u>	<u>Octal</u>
PA0	003	PA1	076	PA2	060
A	040	A1	006	AE	056
AE1	057	AH	044	AH1	025
AW	075	AW1	023	AY	041
B	016	CH	020	D	036
E	054	EH	073	EH1	001
EH2	000	EHH	010	ER	072
F	035	G	034	H	033
I	047	I1	013	I2	012
IE	074	IH	005	IU	066
IY	011	J	032	K	031
L	030	M	014	N	015
NG	024	O	046	O1	065
O2	064	OO	027	OO1	026
OOH	004	P	045	Q	002
R	053	S	037	SH	021
T	052	TH	071	THV	070
U	050	U1	067	UH	063
UH1	062	UH2	061	UH3	043
V	017	W	055	Y	051
Y1	042	Z	022	ZH	007

\*Null Code 277

## APPENDIX B



```
* BN00~BN23 -- Accumulator bus, 24 lines, bi-directional,
               "low" when enabled.
```

\* PS is SPU & T4

\* TC is SPU & T & A

\* OSC -- Clock pulse source, "low" during clock pulse.

## BIBLIOGRAPHY

### Books

- Donovan, John J., Systems Programming, NY, McGraw-Hill Book Co., 1972.
- Flanagan, J. L., Speech Analysis Synthesize and Perception, Berlin-Heidleberg, Springer-Verlag, 1972.
- Knuth, Donald E., The Art of Computer Programming, Vol. 3 (3 volumes), California, Addison-Wesley Publishing Co., 1968.

### Articles

- All About Voice Response, Datapro 70, September, 1974.
- Bulletin #ETB-5-0302-MD, Troy, Michigan, Vocal Interface Division of the Federal Screw Works, 1973.
- Clements, David T., Voice Output for Student Information Inquiry, Costa Mesa, California, Coast Community College District, 1974.
- Mellroy, M. D., Synthetic English Speech by Rule, Murray Hill, New Jersey, Bell Telephone Laboratories, March, 1974.

### Reports

- Fairchild F24 Systems Manual (Hardware), Vol. 3 (3 volumes), California, Fairchild Instrumentation, 1970.

## Unpublished Materials

Scott, Dan W., unpublished card loader program, Department of Computer Sciences, North Texas State University, Denton, Texas.

Scott, Dan W., unpublished F24 simulator on the IBM 360, Department of Computer Sciences, North Texas State University, Denton, Texas.

Scott, Dan W., "The Fairchild F24 Computer Systems Manual," unpublished manual, Department of Computer Sciences, North Texas State University, Denton, Texas, 1973.

Scott, Dan W., unpublished memory dump program, Department of Computer Sciences, North Texas State University, Denton, Texas.

Scott, Dan W., unpublished one-card bootstrap loader, Department of Computer Sciences, North Texas State University, Denton, Texas.

Smith, Russel, unpublished interface design, graduate student, Department of Computer Sciences, North Texas State University, Denton, Texas, 1975.